

IN THE CLAIMS

A listing of the claims of the present application is as follows:

1. (Original) A method of managing storage of objects of sizes smaller than a storage transfer unit in a computer system, comprising the steps of:

maintaining a plurality of storage transfer units in a first storage medium organized by a quantity of free space in a storage transfer unit;

maintaining in a second storage medium a cache comprising a copy of at least one of said plurality of storage transfer units;

in response to a request to store an object of a size less than a storage transfer unit:

searching for a cached storage transfer unit with sufficient free space to store the object;

if no such cached storage transfer unit can be found, identifying an uncached storage transfer unit with sufficient free space to store the object and storing a copy of the identified storage transfer unit in the cache; and

storing the object in the identified storage transfer unit by modifying at least one data structure in the cache and subsequently writing a cached copy of the storage transfer unit to the first storage medium.

2. (Original) The method of claim 1, wherein the first storage medium comprises disk storage.

3. (Original) The method of claim 1, wherein the second storage medium comprises main memory.

4. (Original) The method of claim 1, wherein the step of searching for a cached storage transfer unit further comprises identifying a cached storage transfer unit with sufficient free space to store the object giving preference to such cached storage transfer units with less free space.

5. (Original) The method of claim 4, wherein the step of searching for a cached storage transfer unit further comprises identifying a cached storage transfer unit with a least amount of free space sufficient to store the object.

6. (Original) The method of claim 1, wherein the step of identifying an uncached storage transfer unit further comprises identifying an uncached storage transfer unit with sufficient free space giving preference to storage transfer units which minimize fragmentation.

7. (Original) The method of claim 1, wherein the step of identifying an uncached storage transfer unit further comprises giving preference to storage transfer units with more free space.

8. (Original) The method of claim 7, wherein the step of identifying an uncached storage transfer unit further comprises identifying a storage transfer unit with a most free space.

9. (Original) The method of claim 1, further comprising the steps of:
maintaining at least one list of storage transfer units;
maintaining at least one tail pointer to a plurality of contiguous unallocated storage transfer units;

wherein the step of identifying an uncached storage transfer unit further comprises the steps of searching for an uncached storage transfer unit on the at least one list with sufficient space, and if such an uncached storage transfer unit can not be located, identifying an unallocated storage transfer unit from the at least one tail pointer.

10. (Original) The method of claim 1, further comprising the step of maintaining a plurality of lists of storage transfer units organized by a quantity of free space in a storage transfer unit.

11. (Original) The method of claim 1, further comprising the step of maintaining at least one tail pointer to a plurality of contiguous unallocated storage transfer units.

12. (Original) The method of claim 1, wherein a cached copy of a storage transfer unit is written to the first storage medium in response to at least one of: (i) an object in the storage transfer unit being updated; (ii) a number of changed bytes in the storage transfer unit exceeding a threshold; (iii) a number of changed objects in the storage transfer unit exceeding a threshold; and (iv) the cached copy being about to be purged from the cache.

13. (Original) The method of claim 1, wherein an application program writes at least one storage transfer unit to disk in a transactional manner.

14. (Original) The method of claim 12, wherein the cached copy is about to be purged from the cache as a result of at least one of a cache replacement policy and the computer system being about to go down.

15. (Original) The method of claim 1, wherein a storage transfer unit comprises a sector.

16. (Original) A method of maintaining a plurality of objects in a storage transfer unit, comprising the steps of:

identifying an object position in the storage transfer unit by an object offset in the storage transfer unit;

in response to a request to one of access and update a storage transfer unit, copying the storage transfer unit so that different objects are copied into different buffers;

performing at least one update to at least one object in the storage transfer unit by modifying at least one buffer; and

after the at least one update has occurred, updating the storage transfer unit from the at least one buffer.

17. (Original) The method of claim 16, further comprising the steps of:

when an object which does not have a highest offset is deleted, adding the offset to a list;

satisfying an allocation request by using an offset from the list; and

if an offset is not reused by the time the storage transfer unit is updated from the at least one buffer, storing a placeholder on the storage transfer unit indicating the object has been deleted.

18. (Original) The method of claim 16, further comprising the step of:

in response to an object update which would cause a storage transfer unit to overflow, moving the object to a new storage transfer unit and storing a forwarding pointer in the previous storage transfer unit.

19. (Original) The method of claim 16, wherein the storage transfer unit is maintained on disk and the at least one buffer is maintained in main memory.

20. (Original) The method of claim 16, wherein the step of updating the storage transfer unit from the at least one buffer further comprises copying a plurality of objects from buffers to the storage transfer unit in a contiguous area so that free space in the storage transfer unit is contiguous.

21. (Original) The method of claim 16, wherein the storage transfer unit comprises a sector.

22. (Original) The method of claim 16, further comprising the step of maintaining a number of free bytes in the storage transfer unit.

23. (Original) The method of claim 22, wherein the step of performing at least one update further comprises using the number of free bytes in the storage transfer unit to prevent overflow.

24. (Currently Amended) The method of claim 16, wherein the copy of at least one of the plurality of storage transfer units included in the cache is one of a partial copy and an inexact copy.

25. (Original) Apparatus for managing storage of objects of sizes smaller than a storage transfer unit in a computer system, comprising:

at least one processor operative to: (i) maintain a plurality of storage transfer units in a first storage medium organized by a quantity of free space in a storage transfer unit; (ii) maintain in a second storage medium a cache comprising a copy of at least one of said plurality of storage transfer units; (iii) in response to a request to store an object of a size less than a storage transfer unit: searching for a cached storage transfer unit with sufficient free space to store the object; if no such cached storage transfer unit can be found, identifying an uncached storage transfer unit with sufficient free space to store the object and storing a copy of the identified storage transfer unit in the cache; and storing the object in the identified storage transfer unit by modifying at least one data structure in the cache and subsequently writing a cached copy of the storage transfer unit to the first storage medium.

26. (Original) Apparatus for maintaining a plurality of objects in a storage transfer unit, comprising:

at least one processor operative to: (i) identify an object position in the storage transfer unit by an object offset in the storage transfer unit; (ii) in response to a request to one of access and update a storage transfer unit, copy the storage transfer unit so that different objects are copied into different buffers; (iii) perform at least one update to at least one object in the storage transfer unit by modifying at least one buffer; and (iv) after the at least one update has occurred, update the storage transfer unit from the at least one buffer.

27. (Original) An article of manufacture for managing storage of objects of sizes smaller than a storage transfer unit in a computer system, comprising a machine readable medium containing one or more programs which when executed implement the steps of:

maintaining a plurality of storage transfer units in a first storage medium organized by a quantity of free space in a storage transfer unit;

maintaining in a second storage medium a cache comprising a copy of at least one of said plurality of storage transfer units;

in response to a request to store an object of a size less than a storage transfer unit:

searching for a cached storage transfer unit with sufficient free space to store the object;

if no such cached storage transfer unit can be found, identifying an uncached storage transfer unit with sufficient free space to store the object and storing a copy of the identified storage transfer unit in the cache; and

storing the object in the identified storage transfer unit by modifying at least one data structure in the cache and subsequently writing a cached copy of the storage transfer unit to the first storage medium.

28. (Original) An article of manufacture for maintaining a plurality of objects in a storage transfer unit, comprising a machine readable medium containing one or more programs which when executed implement the steps of:

identifying an object position in the storage transfer unit by an object offset in the storage transfer unit;

in response to a request to one of access and update a storage transfer unit, copying the storage transfer unit so that different objects are copied into different buffers;

performing at least one update to at least one object in the storage transfer unit by modifying at least one buffer; and

after the at least one update has occurred, updating the storage transfer unit from the at least one buffer.